AD-A085 346 OFFICE OF THE UNDER SECRETARY OF DEFENSE FOR RESEARCH--ETC F/G 9/2
DEFENSE COMPUTER RESOURCES TECHNOLOGY PLAN.(U)
JUN 79

UNCLASSIFIED NL

1 OF 1
AD
A085346

END
DATE
FILMED
7-80
DTIC

LEVEL I

ADA085346

DEFENSE COMPUTER RESOURCES TECHNOLOGY PLAN.

June 1979

ADVANCED RESEARCH PROJECTS AGENCY

5 Jun 80

MEMO FOR DIRECTOR, DTIC

SUBJECT: Documents for the DTIC Collection

Would you please add the attached to the DTIC collection. The DARPA Program Manager requests that a statement be added to the abstract of the "Defense Computer Resources Technology Plan" to reflect the existence of ADA047062 and ADA022558.

Thank you for your assistance.

W. L. DeWeese
Administrative
Officer

Encl
"DoD Common High Order . ."
"Defense Computer Resources . ."

THE UNDER SECRETARY OF DEFENSE
WASHINGTON, D.C. 20301

RESEARCH AND
ENGINEERING

18 SEP 1979

The Management Steering Committee for Embedded Computer Resources (MSC-ECR) was established to address the problems which have plagued Defense computer applications and to establish mechanisms which will allow DoD to benefit maximally from advanced computer technology. A well managed DoD-wide technology program to provide necessary technological expertise and tools is critical to the success of the comprehensive policy initiatives now underway, and has my strong support.

This Defense Computer Resources Technology Plan has been prepared with the full cooperation of the three Services, DARPA, DCA and NSA. It has been reviewed and approved by the MSC-ECR and implementation is underway. The Major Program Elements identified in the Plan have been designated USD(R&E) special interest items and are given particular attention in the budget and apportionment process.

I encourage Defense contractors to utilize the Plan as guidance for Independent Research and Development efforts to complement this high priority DoD technology program.

William J Perry

A

See also AD-A047062 + AD-A022558

EXECUTIVE SUMMARY

Problems with DoD software management and acquisition were first documented and appropriately recognized in 1974. Since that time, DoD has moved resolutely to solve them. DoD Directive 5000.29, "Management of Computer Resources in Major Defense Systems," was issued in April 1976. It provided an overall policy framework for computer resources management, emphasizing that software is a major systems component and should receive appropriate attention in the *Defense System Acquisition Review Council* (DSARC) process. DoDD 5000.29 also established the Management Steering Committee for Embedded Computer Resources (MSC-ECR) with broad scope and responsibilities to oversee embedded resources and applications.

The Research and Development Technology Panel was established under the auspices of the Management Steering Committee to provide a common DoD-wide approach to future technology development for embedded computer resources. The R&D panel published the first Defense System Software R&D Technology Plan in September 1977. The plan, which covered the period FY1978 to FY1983, provided for the first time a common structure for all DoD software R&D programs. For each of twelve technology areas, problems and issues were listed, existing and proposed R&D directions described, and recommended funding profiles presented.

This is the second edition of the plan; it was prepared with the full cooperation of the three Services, DARPA, DCA, and NSA. Its revised outline provides a more coherent and meaningful grouping of efforts, clarifies the intent of some subdivisions, and adds new categories. The scope has been expanded to include hardware technology needs. All parts of the text have been considerably expanded to provide more comprehensive descriptions of problems, issues, and R&D directions. In each R&D area, specific DARPA, Service, and OSD initiatives are identified.

In some areas efforts are building up and becoming better focused, but in others present budgets are inadequate to allow meaningful efforts to be started. Examples of developed areas are support for standard high order languages, high order language modernization and convergence (Ada), distributed systems technology, and the development of standard militarized computers. Key areas in which deficiencies still exist include multilevel computer security, error-resistant systems, standardization of input/output and network interfaces, and specification of standard reusable software functions.

This R&D Technology Plan provides for longer range planning and will not be issued annually. Annual reports will be published to indicate how ongoing programs are conforming to the plan, and to provide updated technical and financial information. The FY1978 Annual Report, which summarizes accomplishments during FY1978 and detailed program plans for FY1979 to FY1981, is included as Appendix C to this document.

William E. Carlson
Chairman
R&D Technology Panel

CONTENTS

OBJECTIVES AND PROGRAM MANAGEMENT

A. INTRODUCTION

This plan is the result of four years of intensive analysis and evaluation to determine the causes of DoD computer resources problems and to define a comprehensive R&D program to solve them. It addresses technology needs and opportunities for the computers embedded in major weapons systems--for example, those that guide missiles to their targets, aim guns, control and process the data from radars and other sensor systems, and control high-performance airplanes, some of which are inherently unstable and impossible to control without computer assistance.

The rate of progress in integrated circuits and computer technology is startling. Fifteen years ago, all computers were large and expensive. Today, powerful computers for home use cost only a few hundred dollars, and hand calculators cost as little as five dollars. With Very Large Scale Integrated Circuit technology, advanced computers will cost as little as $5 or $10; every home will have several of them, embedded in everything from washing machines to children's toys. Flexible computer-controlled communications systems and large-scale command and control systems with sophisticated decision aids will make it possible to coordinate the actions of globally dispersed forces and achieve a concentration of resources unprecedented in the history of warfare.

The rate at which computer technology advances is an enormous challenge to DoD managers. The software technology for controlling these inexpensive computers is advancing at a much slower rate, so software development and maintenance costs will constitute an ever increasing share of total system life cycle costs. Most development projects are in the unfortunate position of being the first to use one or more new technologies. Personnel must be retrained constantly to stay up-to-date. Managers would like to avoid such risks, but the cost of being very conservative and using only proven technology is unacceptable; for example, using computer hardware that is only four years out of date will probably double the hardware cost for a given level of performance. On the other hand, if designers are too ambitious, unexpected technical problems may cause large schedule slippages and cost overruns. In some cases, very expensive projects have had to be cancelled without delivering any system to users.

The DoD Management Steering Committee for Embedded Computer Resources (MSC-ECR) was established to provide a policy framework and coordination mechanism to allow DoD to benefit maximally from advanced computer technology. The R&D efforts outlined in the following pages will provide the technological expertise and tools critical to the full success of the comprehensive software policy initiatives now under way.

This plan defines a tightly managed DoD-wide technology program that will facilitate interoperability among Service and agency computer systems. It provides for DoD control of programming language standards and the definition of standard hardware/software interfaces. Hardware technology efforts will be closely coordinated to provide for logistics support in combat areas. Distributed systems R&D will be coordinated to achieve a convergence among network protocols and applications interfaces.. Scientists from NATO countries are participating actively in the language modernization and convergence effort (Ada) to insure that the results are acceptable for NATO systems. Participation by NATO countries in other aspects of this program will be actively sought.

The generic R&D identified in this plan will complement an aggressive program of exploratory and advanced development aimed at specific systems requirements. A major cause of DoD software problems has been the tendency to commit to performance requirements for large systems that can be met only with unproven computer technology. The solution is to identify technology risk areas and explore them in the laboratory or in operational testbeds before freezing system designs.

The President's Reorganization Task Force has called for "accelerated development of, and commitment to, information technology which, though not a goal in and of itself, is a means by which an information-intensive society may achieve its objectives." This plan reflects a major commitment on the part of DoD to aggressively manage the exploitation of advanced computer technology in military systems.

B. BACKGROUND

The failure to aggressively manage the use of computers in weapons systems led to an epidemic of cost, schedule, and reliability problems during the early 1970's. By 1974, the situation had reached crisis proportions; software problems seemed so unique and pressing that they became the focus of numerous task forces and studies. Among the most important of these were the following:

- o Electronics - X, January 1974.
- o DoD Weapon Systems Software Acquisition and Management Study, MITRE, May 1975.

- DoD Weapons Systems Software Management Study, Johns Hopkins Applied Physics Laboratory, June 1975.

- Report of the Operational Software Panel of the Navy Laboratory Computing Committee, September 1975.

- Findings and Recommendations of the Joint Logistics Commanders' Software Reliability Work Group, November 1975.

- Report of the Software Technology R&D Panel of the Navy Laboratory Computing Committee, September 1976.

- Operational Software Management and Development for U.S. Air Force Computer Systems, National Academy of Sciences, 1977.

All these studies reached roughly the same conclusion: The basic principles which characterize sound engineering practice in civil, mechanical, aeronautical, and other engineering disciplines were *not* being applied to computer resources. As a result, the computer's enormous potential to improve our military effectiveness was not being realized; instead, a state of confusion and uncertainty existed, provoking operational personnel to ask whether computers would ever fulfill the lofty promises of the computer scientists. The executive summary of the MITRE report provides the following diagnosis:

- The major contributing factor to weapon systems problems is the lack of discipline and engineering rigor applied to the weapon systems software acquisition activities.

- The current acquisition process does not recognize that the most significant part of a software effort, involving the heaviest expenditures of fiscal and manpower resources, occurs early in the process, before completion of the development, in contrast to hardware acquisition, where the heaviest expenditures occur during production and deployment . . . hardware phasing should take into account uncertainties in the software R&D effort and relationships with software.

- Ignoring life cycle considerations early in the process of defining software has, as an example, caused the late availability of software support facilities and the lack of adequate software maintenance resources for some systems.

- The effect of poor software quality and performance and delayed software availability on total system costs is frequently much greater than the direct costs for the software.

- Consistent practices are lacking for the feedback of management information on software efforts to allow recognition of successful methods and to identify common, costly problem areas in which attention should be focused for greatest leverage.

The Chairman of the Software Reliability Work Group told the Joint Logistics Commanders in May 1975:

> "We simply do not have DoD-wide policies for developing reliable software....We have generated a large number of regulations, directives, and military standards for systems acquisition management. The vast majority of the procedures outlined in these documents are not tailored for software. Software considerations have been added to some of them after the fact, but they are still really hardware oriented. The result is that they conflict with each other, use non-standard terminology, and so forth" (SRWG Report, pp. 28-29).

He described vividly the sorry state of affairs in many R&D projects:

> "We build airplanes first and eventually reach the point of seeing how well they fly. Then, we worry about the avionics. Next, we worry about the computers. Finally, and often many years too late, we begin to be concerned about the software" (SRWG p. 29).

Once these problems were documented and recognized, DoD moved resolutely to solve them. As mentioned above, the MSC-ECR (initially called the Weapon Systems Software Management Steering Committee) was established in December 1974 to bring order and discipline into the software management process. That committee drafted DoD Directive 5000.29, which was issued in April 1976 to provide an overall policy framework for computer resources management. Software was recognized as a major system component and given appropriate emphasis in the Defense System Acquisition Review Council (DSARC) process.

The Research and Development Technology Panel of the MSC-ECR, which assembled this plan, was established in August 1976 to coordinate computer resources R&D activities within the military departments and defense agencies, including both embedded computer resources and general purpose automatic data processing applications. The R&D Technology Panel's charter is included as Appendix A to this plan. The panel's job is to

- o Provide DoD-wide coordination of generic computer technology activities.
- o Establish an orderly process for validating improved techniques and tools and a commitment to package and maintain the best tools so that they are available for DoD software projects.
- o Place high priority on exploratory efforts to determine the implications of new computer system design and architecture technology, so that designs for new systems can be based on experimental evidence rather than conjecture.

The technology recommendations of the Joint Logistics Commanders' Software Reliability Work Group and the other study efforts were consolidated and adopted by the MSC-ECR in August 1976 as this program's baseline technology objectives, and are given as Appendix B to this plan. During the next year, technical approaches to achieving the objectives were evaluated and twelve major R&D areas defined. The R&D areas were intended to be action-oriented, suggesting specific research projects and criteria for evaluating their progress. They were placed into three major categories: Life Cycle Management Technology, System Design Principles and Architectural Standardization, and the Implementation and Maintenance Environment. The initial R&D Technology Plan outlining these R&D areas was published in September 1977.

This edition of the plan refines and extends the September 1977 outline, including a detailed discussion of the issues and opportunities in each R&D area. The three Services, as well as DARPA, DCA, and NSA, were active participants in the development of this plan. The scope has been expanded to include hardware technology needs. Revisions to the R&D area outline have been made to clarify the intent of the original plan and to simplify the assignment of projects to categories. To provide close interaction within DoD, an annual Embedded Technology Planning Conference has been established.

C. OBJECTIVES AND APPROACH

This plan is intended to provide coherent direction and guidance to generic computer technology R&D efforts for a period of at least five years (FY1980 - FY1984). It discusses the technical issues in each R&D area and establishes specific objectives and responsibilities. Program and cost information will be provided in an annual report issued each January to summarize progress to date and plans for the next two fiscal years.

The technology initiatives are now divided into four main categories. Table 1 summarizes the projected budget for each category during FY1979-FY1981. The categories are as follows:

- Development of Life Cycle Management Tools to help DoD and industrial program managers better plan and control the software development process. Specific efforts are as follows: requirements analysis; cost/schedule/quality data collection and analysis; metrics and planning technology; specification, control and configuration management; and the publication of policy and procedure guidance documents.

- Development of advanced System Design and Architecture concepts to improve the reliability, usability, adaptability and cost-effectiveness of defense computer applications. Specific efforts aim to develop technology for error-resistant systems, hardware/software/firmware tradeoffs, distributed systems, and multilevel computer security. Applications testbeds and experimental facilities are needed to refine and evaluate new technology.

- Specification and Development of Standard Software Products for military systems. Efforts here focus on the support of current standard programming languages, including the development of programming tools; the development of a new standard language (called Ada) to modernize and standardize the programming environments used by the three Services and to facilitate NATO interoperability; evaluation and convergence of central processing unit instruction sets and input/output interfaces; and examination of applications systems to find common processing functions that occur frequently enough to justify packaging them as reusable software products.

- Development of advanced Computer Hardware technology to meet unique DoD needs. Efforts in this area are to provide competitive sources for standard militarized computers and computer components; develop new design and fabrication technology that shortens the time required to exploit commercial sector hardware technology advances in militarized computers and in very high performance computers for which DoD is the primary customer; and to simplify field maintenance and logistics support.

These R&D areas address high-priority generic technology problems experienced by the Services and defense agencies. This plan recognizes that organizational constraints and large investments in existing hardware/software systems must be taken into account in planning the introduction of new technology. Each Service and agency must perform the necessary analysis and evaluation for its applications. Major technology outputs of this program will be improved policies and procedure guidance, proven hardware interface standards, and the availability of standard hardware and software products for use by DoD organizations and contractors.

TABLE 1

DEFENSE COMPUTER RESOURCES TECHNOLOGY PROGRAMS

	($ millions) FY 79	FY 80	FY 81
A. LIFE CYCLE MANAGEMENT TOOLS			
A.1 Requirements Analysis	1.8	1.8	2.5
A.2 Cost/Quality Data Collection and Analysis	1.1	.8	1.3
A.3 Metrics and Planning Technology	.8	.9	.8
A.4 Specification, Control and Configuration Management	1.2	1.4	1.4
A.5 Policy and Procedure Guidance Documents	1.3	.9	1.1
AREA TOTAL	6.2	5.8	7.1
B. SYSTEM DESIGN AND ARCHITECTURE			
B.1 Error Resistant Systems	.3	.5	.8
B.2 Hardware/Software/Firmware Tradeoffs	2.5	2.6	3.6
B.3 Distributed Systems	3.1	4.1	4.9
B.4 Multilevel Computer Security	1.8	2.6	5.1
B.5 Applications Testbeds and Experimental Facilities	5.5	6.4	9.1
AREA TOTAL	13.2	16.2	23.5
C. SOFTWARE PRODUCT STANDARDIZATION			
C.1 Support of Present Standard Languages	3.0	2.9	2.4
C.2 New Standard Language Development (Ada)	2.2	4.3	5.3
C.3 Specifications of Reusable Software Functions	.5	.5	.6
C.4 Standard Instruction Sets and I/O Interfaces	.8	1.1	1.4
AREA TOTAL	6.5	8.8	9.7
D. HARDWARE			
D.1 Standard Military Computers	5.6	14.0	18.5
D.2 Standard Military Peripherals	.7	4.0	6.0
D.3 High Performance Computers	1.5	1.0	2.0
AREA TOTAL	7.8	19.0	26.5
TOTALS	33.7	49.8	66.8

D. TECHNICAL COORDINATION AND MANAGEMENT

The Defense Computer Resources Technology Program requires tight coordination of Service and DARPA efforts. Existing procedures of the DoD budget cycle are adequate to insure program coordination. The R&D Technology Panel and an annual Embedded Computer Technology Planning Conference will be the primary mechanisms for achieving detailed technical coordination. As mentioned above, an annual report published each January will evaluate progress to date, identify problems requiring special emphasis in the future, and provide detailed program planning data. Software tools developed in this program will be installed in the National Software Works (NSW) for evaluation. When the NSW is complete, it will become the primary repository and distribution mechanism for DoD's software technology and tools. The remainder of this section discusses these management and technical coordination mechanisms in greater detail.

The following program elements will be monitored to insure that adequate funds are provided for projects supporting this plan:

Army:	62701A
	62725A
	62746A
	63723A
	65803A
Navy:	62721N
	63526N
	64501N
Air Force:	62204F
	62702F
	63728F
	64740F
DARPA:	62708E
DCA:	33126K

In addition, basic computer science research with a potential to improve substantially the use of computers in DoD applications will be given strong support by the Services and DARPA. As promising new ideas are produced in the 6.1 program elements, they will be transitioned into the program elements listed above for refinement and exploitation.

Major events in the annual coordination cycle are as follows:

October

R&D Technology Panel meets to discuss program progress and to begin compiling the annual report.

January

The Chairman of the R&D Technology Panel presents the annual report for the previous fiscal year to the MSC-ECR.

April

Each significant project submits appropriate technical papers to the R&D Panel for review and inclusion in the annual Embedded Computer Technology Conference.

The R&D Panel meets to discuss program progress and to review the papers.

June

The Embedded Computer Technology Planning Conference is held.

July

OUSDR&E (R&AT) meets with Services to discuss apportionment review issues.

August

MSC-ECR is briefed on the resolution of apportionment review issues and the plans for the coming fiscal year.

The Embedded Computer Technology Planning Conference will provide a forum for technologists and users to share ideas, talk about the practicality of the efforts in progress, and discuss future R&D directions. Presentations on advanced system concepts will be included to provide a bridge between this program's generic R&D and R&D for specific applications.

The proceedings of the Embedded Computer Technology Planning Conference will be the primary document for disseminating results of this program to DoD organizations and contractors. Technical papers will cover main technical thrusts of each project rather than focusing on isolated issues; they will also contain extensive bibliographies listing project reports. The papers should be written by the person actually doing the work, so appropriate provisions for this requirement should be included in software R&D contracts. The intended audience consists of software engineers and technical personnel rather than program administrators. In addition, the R&D panel will select one or more technical areas in which major progress has been achieved and organize sessions at national computer conferences to make defense contractors aware of the new technology.

All reports for the Defense Computer Resources Technology Program will be submitted to the Defense Documentation Center, and DDC order numbers will be included in the bibliography of the Planning Conference proceedings.

The annual report published in January will satisfy the need for a succinct management summary of program progress. It will include summary reports and recommendations prepared by Embedded Computer Technology Planning Conference session chairmen. Achievements during the past fiscal year will be highlighted. The report will summarize actual expenditures during the last fiscal year, detailed plans for the current and two succeeding fiscal years, and predicted directions for the outyears.

Industrially funded organizations will report management and technical planning expenses which cover several or all R&D areas under area A.5, Policy and Procedure Guidance Documents. The outputs of these management activities will include technical reports which summarize how the generic computer technology being developed under this plan is expected to impact future embedded computer resources development and maintenance activities.

Tools developed under this program will be installed in the National Software Works to facilitate their distribution and evaluation. The NSW provides a coherent user interface to tools running on several different computers, and is accessible at most military laboratories and R&D centers via the ARPANET. The Air Force will develop a detailed plan for establishing a fully operational version of NSW to serve as the DoD software tool repository and testbed for experimental tools. The plan will specifically address necessary protocol modifications to achieve interoperability with the Navy Laboratory Computer Network (NALCON), the Air Force Systems Command Scientific and Management Network (AFSCNET), and AUTODIN II. The plan will also specify how military and commercial circuits can be used to provide cost-effective access to the NSW from any DoD or contractor site in the CONUS. The existing experimental NSW system should be used as an interim tool repository during FY1979 - FY1981, with a transition to a fully operational NSW by FY1982. Applications testbeds and experimental facilities will be made interoperable with the NSW to the maximum extent practical.

TECHNOLOGY AREA SUMMARIES

A. LIFE CYCLE MANAGEMENT TOOLS

A.1 Requirements Analysis

Requirements analysis is the process of balancing needs and desired capabilities against technical risk and cost to define a coherent, practical package of capabilities. It provides for the R&D of methodologies and supporting software tools for computer resource requirements analysis, including technology for modeling systems conceptualized by their requirements, tracking these requirements as they evolve over the system life cycle, and for relating requirements to system design specifications.

Problem Summary

- o Delivery of unacceptable systems to users.
- o Schedule delays and cost growths attributable to requirements that can be described by any or all of the following pejorative adjectives: excessive, infeasible, incomplete, conflicting, ambiguous, inconsistent, untraceable, and changing.
- o Delays in exploiting new computer technology in defense systems.

Technical Issues and Approach

- o Studies of DoD software problems have been unanimous in their indictment of the requirements definition process for defense computer systems.
- o Technology is needed that will allow DoD components to develop, analyze, and manage requirements effectively throughout the system life cycle. Specifically, the following issues must be addressed:
  - What to include (structure, data flows, program control).
  - Level of detail controlled by the government.

- Level of detail maintained by the contractor.
- Manner of presentation (text, graphic, computer-aided).
- Relationship between the requirements definition process and the design process.
- Constraints imposed by external interfaces.
- Audit trail on requirements evolution.
- Change impact assessment.
- Balancing need against cost and risk.
- Bookkeeping systems for correlating requirements with designs at a detailed level.
- How to analyze requirements for completeness and consistency.
- The role of models, demonstrations, and prototypes.

o One cause of requirements definition problems has been a failure to recognize explicitly the evolutionary nature of defense systems, most particularly software-intensive systems. The acquisition goal of delivering a completed system on the first try is unworkable for many reasons, among them the following:

- The threat environment changes constantly.
- Organizational responsibilities and priorities change.
- Users lack experience with automated systems, so their perceived needs change as they learn what the technology can do for them.

o A large fraction of life cycle costs is charged to software maintenance. That software is easier to modify than hardware is a major advantage of software technology. The ability to exploit the inherent modifiability of software to adapt to a changing environment, and at the same time to control costs and insure the reliability of operational software, will be a prerequisite to victory in future military conflicts and other international competition. Hence, the technological issues involved in managing requirements evolution must be a major focus of R&D in requirements analysis.

- Requirements analysis R&D must take into account the impact of new system design and architecture technology, and changes in the implementation/maintenance environment. For example:

  - A bookkeeping tool for correlating requirements with designs should interface to the tool that correlates the design with the implementation.

  - A thorough understanding of technology options and tradeoffs is a prerequisite to successful requirements analysis, so testbeds will play a key role in requirements risk analysis.

  - Languages for the rigorous specification of hardware and software interfaces may be ideal for specifying external interface requirements.

Research Direction and Action

- The Air Force will lead the development of requirements analysis technology to meet these needs. Particular attention will be given to the issue of risk identification and reduction during the concept definition and validation phases.

- All three Services will evaluate new requirements analysis aids as they are available.

- DARPA will develop advanced technology for expressing requirements in a form which is both computer-processable and easy for people to understand. A mixture of natural language and graphic techniques will be explored.

A.2 Cost/Schedule/Quality Data Collection and Analysis

This R&D area provides for collecting quantitative cost and quality data from ongoing systems (of all Services) and identifying major influence factors, the specification of data to be collected, and the precise definition of measurement techniques. Use of this data for cost and schedule estimation is covered in A.3; use for evaluation of techniques and establishment of guidelines is covered in A.5.

Problem Summary

- o Lack of quantitative cost/quality metrics.
- o Lack of historical data as basis for cost/schedule/performance prediction.
- o Lack of historical data as basis for evaluating impact of new technology.

Technical Issues and Approach

- o An understanding of how various factors influence cost, schedule, and quality is needed as a basis for defining design constraints and as a guide to test and evaluation strategy.
- o Comparison of analogous systems is a technique used for cost estimation and sizing in most engineering disciplines. DoD's investment in this R&D area will focus on collecting representative data about defense computer applications, including qualitative as well as quantitative information, to provide a basis for such comparative judgments. An example of qualitative data is a project history file showing major decisions, key personnel changes, etc.
- o Specific issues which must be addressed include:
  - Measures of software team productivity.
  - Measures of computer resource reliability.
  - Measures of effectiveness for software validation and certification.
  - Measures of software efficiency.
  - Measures of structural integrity for software that indicate the feasibility of future modification and enhancement.
  - Measures of effectiveness for evaluating computer systems in specific applications.
- o Research to address these issues is impossible without adequate data on computer systems and projects; hence, historical data on DoD projects is an important resource which should be collected in a repository and organized for easy access.

Research Direction and Action

- The Air Force will operate the repository for computer resources cost, schedule, and quality data.

A.3 Metrics and Planning Technology

This R&D area includes the development of life cycle (including operation and support phases) cost estimation, sizing and scheduling models, and the development of objective criteria for determining the successful completion of software design, development, implementation, and test milestones. Empirical data is obtained from the software cost and quality data collection activities (described in A.2).

Problem Summary

- Inaccurate cost/schedule/performance projections.
- Inadequate visibility into and control of the development process.
- Subjective cost/quality performance criteria.
- Lack of operation/support planning and control.

Technical Issues and Approach

- DoD acquisition policy emphasizes the need for risk assessment and minimization during design validation. Unfortunately, the technology for estimating risk and for validating computer software designs is inadequate.
- The current approach is to make design information visible and manage a software development with interim sets of products/reviews oriented toward configuration management. These call for a complete and highly detailed description of system design attributes before the contractor can code and test his design. The net effect is to force the contractor to design software in a controlled environment, where iteration of the design can occur only through the formally documented engineering change process. Many people believe that this rigid control over the design before it has been proved to work is a major cause of software problems.
- Problems in the software acquisition process are sometimes camouflaged by highly visible progress in the acquisition and deployment of computer hardware.

- o Problems are frequently encountered in synchronizing hardware and software acquisition.

  - Selecting hardware too early may constrain the design in undesirable ways.

  - Ignoring hardware constraints early in the design process can allow the pursuit of unrealistic requirements and impractical designs.

  - While emulation and simulation have proven useful in some cases, additional analysis and pilot projects still have a great potential to improve the acquisition process.

- o Commercial software vendors typically plan to release new versions of their software products every six to twelve months. Explicit planning and budgeting for future releases make it easier to decide what capabilities to include in the current version. DoD, on the other hand, usually plans the initial system development as if the software were going to be frozen after the initial delivery to users. Once the system is delivered to the maintenance organization, however, a process of periodic enhancement takes place which is very similar to the process for commercial software products.

Research Direction and Action

- o A Computer Resources Acquisition Panel of the MSC-ECR will be formed to lead the development of improved computer resource acquisition models. It is anticipated that the improved acquisition process will

  - Use system versions, each of which performs end use system functional capabilities.

  - Make cost and schedule uncertainties explicit, rather than committing to unrealistic estimates.

  - Deal with technical risk by building explicit technology development and engineering development phases into the process.

  - Use the products of successful technology R&D programs as the basis for incremental development of operational systems, rather than starting from scratch.

- Include performance optimization phases.

- Consider innovative contractual incentives for reaching agreed-upon objectives.

o All three Services will support the activities of the Computer Resources Acquisition Panel.

o The Air Force will define and document proposed acquisition models.

o The Navy will develop techniques for estimating hardware and software performance requirements and logistics support costs.

o The Army will develop statistical techniques for estimating the cost of software development.

o All DoD components will evaluate the effectiveness of computer resource acquisition models and cost estimation techniques for their applications.

A.4 Specification, Control, and Configuration Management

This R&D area is directed at software configuration management and control. It includes configuration item definition, change impact asessment, cost/quality traceability, and interface control.

Problem Summary

o Inadequate interface management.

o Inadequate documentation.

o Inconsistent application of configuration item control and accounting procedures.

o Inadequate cost/quality traceability.

o Nonrigorous change control.

Technical Issues and Approach

o The problems listed above are largely managerial and will be addressed by the MSC-ECR and the Acquisition Panel of the MSC-ECR. However, many of the managerial problems would be simplified if technology were available to automatically verify the consistency of interface specifications and to reduce the administrative burden of maintaining required documentation.

- o Key technology needs are the following:

  - A sound technical basis for configuration management policy.

  - Simplifying configuration management techniques as much as possible.

  - Evaluating the cost-effectiveness of alternative configuration management techniques.

  - Product definition for configuration management tools suitable for GFE or GFP distribution.

  - Defining the interface between the configuration management system and

    - hardware configuration

    - requirements definition aids

    - planning and cost estimation aids

    - cost accounting systems (which must be unique to each organization)

    - software validation technology

    - programming environment

  - Criteria for regular performance and error monitoring.

- o Changes to software should be incremental rather than continual. Every change has the potential for error, and for causing unreliability in operation, so thorough validation is required. To minimize the cost of validation, and to insure that untested code is never delivered to users, changes must be grouped together into releases and rigorous control must be maintained over the verification, certification and release process.

- Technical and administrative management of changes is often overly optimistic, ignoring the complications of proposed changes. The simplicity of the physical actions involved in modifying software has led to the mistaken belief that modification of software is easy. Resorting to invariant software substitutes (digital hardware and firmware) is one way to enforce an incremental change discipline, but the cost is prohibitive. Traditional software provides the best and least expensive means to accomplish changes, and there is no reason why the necessary management discipline cannot be maintained.

- Requirements change over the life of a system and must be accommodated by well-planned incremental changes to software. The management controls and documentation standards that insure configuration management discipline must not introduce excessive costs and delays into the software modification process.

- Responsibility for a system may shift from one organization to another several times during the system life cycle. For example, the system may be specified by an in-house laboratory, developed by a contractor, maintained by a second in-house organization, and modified to a major extent by another contractor.

Research Direction and Action

- The Computer Resources Acquisition Panel of the MSC-ECR will coordinate the DoD attack on the management and technological issues listed above.

- The Air Force will carefully review existing configuration management policies for consistency and technical practicality. The policy impact of proposed new software engineering methods and tools will be documented. A thorough evaluation of existing configuration management systems will be accomplished.

- DARPA will work with the Air Force to develop automated configuration management aids for geographically distributed systems and to determine the implications for configuration management of multiple systems interoperability.

- The Army, Navy, and DCA will evaluate and use the Air Force and DARPA products.

A.5 Policy and Procedure Guidance Documents

This R&D area provides for the publication of software acquisition management guidebooks which provide a collection of "lessons learned" and discuss the implications of decision options and alternatives. It also includes R&D management activities in support of this plan which cover several or all R&D areas. *This R&D area is the principal bridge* between the research community and the day-to-day world of program managers, system project offices, contracting officials, and contractors.

Problem Summary

- Lack of systems engineering methodology and discipline.
- Lack of technology transfer from R&D into application.
- Lack of a formal process for evaluating and authorizing use of new technology.
- Insufficient understanding by manager and insufficient technical support for the manager.
- Lack of skill continuity over life cycle.
- Personnel obsolescence.

Technical Issues and Approach

- Most reasonable practitioners believe software development is controllable. The essential elements of successful management are recognized: use of proven software engineering techniques, well chosen programming tools, workflow organization, substantive reports and qualified customer reviews, and realistic cost and schedule allowances. Nonetheless, the various studies of DoD software problems indicate that proven tools and techniques are often not applied.
- Guidance documents (e.g., guidebooks, specifications, standards, etc.) have been prepared over the last two years to summarize the best proven software engineering techniques and tools. Evaluation of existing guidance and distillation to focus on major issues, high level professional consensus, and continuous training in the use of those techniques are needed.

- Existing guidebooks cover a variety of topics: Regulations, Specification, and Standards; Contracting for Software Acquisition; Monitoring and Reporting Software Development Status; Statement of Work Preparation; Reviews and Audits; Configuration Management; Requirements Specification; Software Documentation Requirements; Verification; Validation and Certification; Software Maintenance; Software Quality Assurance; Software Cost Estimating and Measuring; Software Development and Maintenance Facilities; Life Cycle Events; and Series Overview.

- Guidance by system program offices must be evaluated to identify strengths, weaknesses, ambiguities, and needs for clarifications and additions.

- Specific acquisitions must be reviewed to relate "what is" to "what ought to be."

- Guidance for system and software managers during the operations and maintenance phase must be expanded. This portion of the life cycle, often representing more than 50 percent of total costs, has been largely overlooked in the guidance.

- New technology must be evaluated, its impact on the acquisition process determined, and the policies and procedures guidebooks updated to encourage the use of the new techniques and tools.

Research Direction and Action

- This R&D area will include management and technical planning expenses which cover several or all R&D areas. The outputs of these management activities will include technical reports which summarize how the generic computer technology being developed under this plan is expected to impact future embedded computer resources development and maintenance activities.

- The Computer Resources Acquisition Panel of the MSC-ECR will coordinate the periodic updating of the guidebooks to reflect and encourage the use of new technology. This activity will require a small amount of R&D funding to evaluate evidence supporting the use of new technology and to identify required changes in policies and procedures.

- The guidebooks will be used for personnel development and training (but no R&D funds will be needed to support such activities.

B. SYSTEM DESIGN AND ARCHITECTURE

B.1 Error-Resistant Systems

This R&D area includes technology for formal verification, fault localization, fault recovery, fault elimination, quality assurance, and associated development methodologies.

Problem Summary

- o Lack of effective design principles for using additional hardware to improve reliability.

- o Lack of system optimization with respect to both hardware and software.

Technical Issues and Approach

- o Examples of DoD applications which require ultra-high reliability are nuclear weapons control, avionics, space and flight control.

- o It is theoretically impossible to test the response of operational software to every possible input.

- o Errors and "system crashes" are a fact of life for users of large systems. For example, 18 software errors were discovered during the ten-day flight of Apollo 14 despite one of the most thorough testing programs that software has ever been subjected to.

- o Research in fault-tolerant systems has focused on detection and recovery from hardware errors. Latent software errors are now the major cause of system unreliability. Furthermore, inadequate software technology is the primary impediment to increased use of hardware redundancy to improve overall computer system reliability.

- o If we knew what algorithms to implement, there would be no major economic or physical barriers to the use of additional hardware to improve total system reliability in most DoD computer applications. Semiconductor logic densities continue to improve by a factor of four every 3-5 years, and both the cost and size of computer hardware generally follow that trend.

- Improved tools for formally specifying and verifying properties of computer software are the key to progress in this area. It will not be necessary to verify every line of code in large systems. System designers can organize software to minimize the number of lines of code that affect critical failure modes. Verification tools will be used to verify the relevant code against specific failure patterns.

  - Software verification tools should be useful for deciding what error detection and correction mechanisms to incorporate into the hardware.

  - Software verification tools should also provide a criterion for evaluating the usefulness of run-time error checks.

- Once the basic theory of error-resistant systems is established, engineering tradeoffs associated with specific DoD applications must be taken into account.

Research Direction and Action

- DARPA has the lead in developing formal verification technology and will also develop techniques for using multiple processors running different instruction sequences to improve reliability.

- The Air Force will take the lead in investigating error-resistant computing techniques for space and avionics applications, the Navy for shipboard and undersea applications, and the Army for ground tactical systems. Section B.4 below provides for the development of "fail-safe" computer systems to protect classified information, and will draw upon error-resistant systems technology.

B.2 Hardware/Software/Firmware Tradeoffs

This R&D area focuses on the software technology implications of innovative digital hardware technology, including Very Large Scale Integrated Circuits, language machines, microprocessors, multi-processor architectures which tightly couple large numbers of processors, associative memories, and advanced archival memory systems.

Problem Summary

- o Proliferation of special-purpose computers built with microprocessors.
- o Lack of system optimization with respect to both hardware and software.
- o Exploiting state-of-the-art advances without disrupting logistics support.
- o Determining software technology implications of plausible hardware technology possibilities, so that the necessary software technology base is available to system developers as soon as the hardware is available.

Technical Issues and Approach

- o The density of semiconductor logic improves by a factor of four every three to five years. This unprecedented rate of technological innovation leads to corresponding improvements in the cost, performance, and reliability of embedded computer circuitry. The improvements, however, do not apply equally to all computer components. Hence, engineering "rules of thumb" must be reviewed constantly and revised to fit current and next generation hardware price/performance characteristics.

- o Two examples will clarify these issues.

  - The first is the problem of choosing the correct primary memory size for an application. The finite size of primary memory is a major constraint on software design; memory maps are key system design documents. Yet memory is the easiest area in which to exploit semiconductor chip density improvements. The amount of memory that can be installed in a given space for a specified price should increase by two orders of magnitude during a twenty-year system life cycle.

  - Microprocessors raise an even more complicated set of technical and managerial issues. Microprocessors are "computers on a chip." Because of their low cost and desirable performance characteristics, they are proliferating rapidly in defense systems. It is noteworthy that microprocessors and their associated software are replacing analog circuits in a variety of process control applications at a time when the opposite goal of replacing software with hardware is becoming increasingly popular in DoD. Microprocessors are already having an important impact by allowing clean modular interfaces between software subsystems to be established. The earliest microprocessors introduced some new software problems because of their primitive four-bit and eight-bit instruction sets. Sixteen-bit microprocessors are becoming available that can utilize standard DoD software, and this will greatly increase their applicability.

- o This R&D area will provide embedded systems developers with the information they need to cope with and exploit this rapid rate of progress in digital hardware technology by

  - Determining the software technology implications of plausible hardware technology possibilities 5-10 years in the future.

  - Suggesting desirable input/output interfaces for new digital system components which replace software with hardware.

- o Major R&D areas are

  - Performance prediction and modeling tools. The key is to have very flexible tools for modeling system structures and for exploring their behavior over wide ranges of possible parameters. The goal is not extreme accuracy, since the device parameters are themselves rough estimates; instead, it is to identify the most promising architectures and the key performance bottlenecks for each so that exploratory R&D efforts can be focused appropriately.

  - Specification and evaluation of innovative hardware/software interfaces. One example is a direct high order language execution machine. Another is the use of multiple microprocessor configurations to provide high bandwidth processing for space and avionic applications while masking the multiprocessing complexity from applications programmers.

  - Design and fabrication methodologies/tools, especially for custom Very Large Scale Integrated Circuit chips with low production run volumes.

  - Tools to aid firmware development, validation, and maintenance.

Research Direction and Action

- o The Navy and Air Force will develop tools for predicting the performance of systems that use innovative hardware technology.

- o The Air Force and Navy will develop and test innovative hardware/software interfaces.

- o The Army will evaluate the feasibility of offloading data management functions into separate data base machines.

B.3 Distributed Systems

This R&D area addresses technology for three kinds of distributed systems:

1. Command-level C3I systems, characterized by wide geographic distribution, substantial autonomy of nodes, evolutionary developr nt building on a backbone communications system. An example is the Wc.ıd Wide Military Command and Control System (WWMCCS).

2. Local area C3I systems, characterized by tightly coupled applications software, often identical copies of the same hardware/software system at every network node, may have high bandwidth communication via coaxial cable or optical bus, or lower bandwidth radio communication. An example is the Navy Tactical Data System (NTDS).

3. Real-time control systems, such as avionics and flight control, which are functionally partitioned with dedicated processors for specific sensor and actuator subsystems. A major goal is complexity reduction to minimize R&D and maintenance costs. An example is the F-16 digital avionics.

Problem Summary

- o Inadequate interface standards and protocols for achieving interoperability across vendor product lines, or between applications systems that were not designed originally to communicate with each other.

- o Lack of software technology that DoD needs in several areas (e.g., operating systems, task/resource synchronization, fault detection/ isolation/recovery techniques, protocols for interoperability, etc.) to capitalize on hardware and architecture opportunities.

Technical Issues and Approach

- o Improvements in computer networking technology and reductions in computer hardware cost and size are making distributed processing an increasingly attractive alternative for DoD Command, Control, Communication, and Intelligence (C3I) systems.

- o Potential benefits of distributed processing for C3I include:

  - Substantially improved survivability, since the enemy is presented with many low-value geographically distributed targets instead of a few high-value ones.

  - Improved reliability, since the operational impact of individual hardware failures is greatly reduced.

  - A reduction in communications bandwidth requirements.

  - A computer system architecture that closely models the *organization it supports, thereby eliminating* many of the management problems associated with the current highly centralized approach. Specifically,

    1. Simplified requirements definition, because it is decentralized and the designers/implementers are closer to the users.

    2. Simplified resource management, because each operational unit has control over its own computer resources. Tight coupling of hardware resources to specific operational requirements would also make it much easier to determine the cost/benefit ratio for proposed hardware upgrades.

- o The Digital Avionics Information System (DAIS) project has demonstrated a fully modular architecture for avionics. As a result, the Air Force has defined the 1553B bus interface standard. The suitability of the 1553B standard for other real-time control applications and its relationship to local area C3I network interfaces need to be determined.

Research Direction and Action

- o Each Service and agency will investigate the performance characteristics of network protocols for its applications.

- o DARPA and the Air Force will develop jointly a National Software Works (NSW) on the ARPANET to address technology problems of command level C3I systems. The NSW concept demonstration will specifically address:

  - Distribution of tool kits to support approved high order languages.

  - Interoperability in a heterogeneous (multi-vendor) hardware/operating system environment.

  - Remote software maintenance, distribution, and error diagnosis over a packet communication network.

  - Configuration management of the software for geographically distributed systems.

  - Technology implications of an evolutionary software development strategy in a distributed network environment.

- o DARPA, the Air Force, and the Navy will develop the technology for distributed files and database management in command level and local area C3I systems. The goal is to experimentally determine the engineering tradeoffs among survivability, local autonomy, interoperability, and performance.

- o The Air Force and Navy will refine the technology for distributed real-time control systems, with the Air Force focusing its attention on avionics and flight control applications.

- o DARPA will take the lead in developing a basic theory of distributed systems, including both software and network protocol issues. The purpose of this thrust is to clarify the common technical principles that cut across all distributed applications, and the engineering tradeoffs that lead to different implementations in different applications environments.

B.4 Multilevel Computer Security

This R&D area aims to create technology for controlled information sharing and electronic message exchange between users operating at different security levels.

Problem Summary

- Cost of maintaining physically separate processing facilities for each level of security classification.

- Communication barriers between intelligence analysts and decision makers operating in different security compartments.

Technical Issues and Approach

- At present, there are only two ways to maintain the security of classified information in a computer:

    - Dedicate the computer to processing information at a single security level (system high operation).

    - Operate the computer at different security levels at different times of the day, allowing ample time for erasing all classified information when the system is shifted from a higher classification level to a lower one (periods processing).

- These methods waste computer hardware, most obviously because of the cost and lost availability time for periods processing sanitization, but also because capacity cannot easily be shifted from one level where it is excess to another where it is needed. Hardware cost forecasts suggest that by 1990 it will be cost-effective to procure enough hardware that availability is not an issue, but during the 1980's these hardware costs will still be of concern.

- The most important requirements for multilevel secure computer systems, however, are driven by operational effectiveness rather than cost. Decisions rarely if ever are made using information from just one security level. The highly classified information provides specific details or reinforcement for decisions which draw heavily on background information that is much more widely available. Artificially raising all the supporting information to the highest classification that will be used in the decision process has numerous bad effects:

- It increases the likelihood of bad decisions due to inconsistencies between the databases at different classification levels.

- Without multilevel computer systems, the information must be carried from the lower classification system to the higher on some removable media such as magnetic tape, paper tape, or cards. This manual transfer introduces delays and increases the probability of error. Most important, it is the version of the data on the highly classified system (which is presumably the version on which decisions will be based) that is most likely to be obsolete or otherwise in error.

- Even more serious operational delays occur once the decisions are made and must be implemented. While the plan will reside at the highest classification level, it must be executed by people who lack the appropriate clearances. Hence, as much of the plan as possible must be downgraded and transferred to more widely available computers. If multilevel computer security capabilities to keep track of the classification levels of the various pieces of information that comprise the plan are lacking, a person must manually review the data and downgrade it appropriately. This operation must be performed when time is of the essence.

- Lack of multilevel security capabilities also makes it impossible to automatically compare new intelligence data with the operational databases. The intelligence reports will be at a different classification level than the operational databases. One could copy the operational data into the intelligence system, but that opens the door to all the problems of error and obsolescence discussed above.

o This R&D area is closely related to area B.3, Distributed Systems, since the operational issues discussed above become much more serious when users in different locations (or at least different rooms) access the system through on-line interactive terminals. A multilevel secure system is also a special type of error resistant system.

o Research in the specific area of multilevel security will focus on

  - Definition of multilevel computer security: what risks are acceptable and what risks are not.

  - Certification problem: how one can evaluate whether a system provides a specified level of security.

- Demonstrations: implement a sequence of prototype systems with increasing levels of performance and safety.

- Technology Building Blocks: create and demonstrate new hardware/software technology which overcomes key performance bottlenecks. For example, process authentication and the cost of switching protection domains are key limitations at present.

Research Direction and Action

- *OASD(C3I) will coordinate tri-Service and agency* efforts to solve the problems discussed above.

- The Services, DCA, and NSA will focus on refining the definition of multilevel security, developing certification techniques, and developing prototype secure systems.

- DARPA will have the lead in developing new technology building blocks and architectural concepts.

B.5 Applications Testbeds and Experimental Facilities

Testbeds are an essential component of the strategy for transferring new computer technology into operational use. Testbeds aim to model the essential aspects of the operational environment, creating a context in which a variety of system concepts can be evaluated and refined. This breadth distinguishes testbeds from prototypes, which are typically built to validate a single system concept.

Problem Summary

- Lack of insight into human factors aspects of user requirements.

- Lack of insight into design alternatives.

- Lack of tools to facilitate design tradeoffs.

- Lack of facility for technology demonstration and transfer.

Technical Issues and Approach

- o Because relatively few systems are built from scratch, the only way to have an impact is to make new technology fit the constraints of the existing systems environment. On the other hand, technological changes which adhere to all the constraints implicit in the existing operational system are unlikely to have much impact.

- o Environments are needed in which technologists can experiment with major modifications to existing systems. These testbed environments will allow technologists to show the benefits of new technology and the feasibility of injecting it into operational environments without disrupting vital functions.

Research Direction and Action

Testbeds are tied to specific operational environments and constraints. They are typically mission-funded, in whole or in part. The Teleprocessing Design Center at Ft. Monmouth, N. J., and the RADC Experimental Facility are used primarily in support of the generic computer technology program described in this plan. The following are other major DoD testbed activities demonstrating new computer technolgies:

- o DCA is developing a WWMCCS Testbed at the Command and Control Technical Center and a Network Testbed at the Defense Communications Engineering Center.

- o DARPA and the Navy are developing the Advanced Command and Control Architectural Testbed on the ARPANET.

- o DARPA and the Army are developing the Army Data Distribution System Testbed at Ft. Bragg, N. C.

- o The Air Force has an Avionics System Analysis and Integration Laboratory (AVSAIL) at the Air Force Avionics Laboratory.

- o The Navy has a Basic Avionics Systems Integration Concept Testbed (BASIC) at the Naval Air Development Center.

C. SOFTWARE PRODUCT SPECIFICATION AND STANDARDIZATION

C.1 Support of Present Standard Languages

This R&D area provides for establishing language control procedures, compiler validation tools, compiler technology enhancement, and the development of a full repertoire of programmer tools for each approved high order language.

Problem/Issue Summary

- o Lack of language standardization and control.
- o Lack of transferability of tools from one project to the next.
- o Lack of rigor and discipline in software development/maintenance.
- o Lack of explicit decision process for introducing new technology.

Technical Issues and Approach

- o Industry has developed tools and methodologies that can significantly improve the software development process. The tools available for DoD languages and militarized computers have generally been of lower quality than those available for the most popular computers in the commercial marketplace. A comprehensive integrated set of tools and methodologies should be available for use on all major DoD software acquisitions regardless of prime contractor.

- o DoDI 5000.31 approves an interim list of seven languages for use in major defense systems. Two of the languages, FORTRAN and COBOL, are widely used for commercial applications. The FORTRAN and COBOL language specifications are controlled by the American National Standards Institute (ANSI), compilers are available for almost all vendors' computers, and programmers using those languages have available to them a large collection of existing software tools. Of the other five languages, only CMS-2 could be described as having an adequate support infrastructure at the time DoDI 5000.31 was issued.

- o The reason J3, J73, TACPOL and SPL-1 were not fully supported is, of course, the proliferation of languages that existed prior to 5000.31. With every project inventing its own language, there was neither the money nor the incentive to do an adequate job of supporting any particular language.

- o DoDI 5000.31 stopped the proliferation. Now the support environment deficiencies must be eliminated for approved standard languages.

Research Direction and Action

The Air Force is responsible for J73 and J3, the Navy for CMS-2 and SPL-1, and the Army for TACPOL. The responsible organization for each language will develop a language control and support program, which will address the complete tool environment for the languages, including

Compilers and compiler validators
Standards checkers
Structured code preprocessors and analyzers
Program development support libraries
Code verification tools
Software test case generators

C.2 New Standard Language and Associated Programming Tools (Ada)

This R&D area provides for the development of a common high order language for DoD-wide use in embedded computer applications, establishment of a control capability for the new language, including computer validation tools, and development of an initial set of compilers and programming tools. The language will be named Ada.

Problem Summary

- o Lack of state-of-the-art tools for military computers and programming languages.

- o Lack of explicit decision process for introducing new technology.

- o Lack of language and tool compatibility on programs involving more than one Service/agency.

- o Lack of transferability of tools from one project to the next.

Technical Issues and Approach

- o Currently approved high order languages were designed for specific applications and/or specific computer hardware. Recent programming language concepts make feasible the development of a single language that meets the needs of almost all DoD applications.

- o High order system implementation languages are proliferating the way business-oriented programming languages proliferated in the late 1950's. The technology is ripe, and in the absence of a national standard the vendors are moving rapidly to provide their own incompatible product offerings. In the 1950's DoD took the lead in the development of the COBOL standard. Given the impact that a common systems implementation language would have on embedded computer systems, it is appropriate that DoD once again take the lead in developing a U.S. and NATO standard.

- o Any common language would provide significant benefits, among which are the following:

  - Joint Service and NATO interoperability.

  - Improved transfer of tools from one project to the next.

  - Simplified cross-training of personnel.

  - Improved focus, and hence productivity, from software R&D that depends on the language: e.g., compilers, verifiers, run-time libraries, operating systems, data management systems, communications protocols, etc.

  - Easier exploitation of hardware technology innovations, since it is much easier to transfer applications software to new generation hardware.

- o Greater benefits can be obtained by defining a new common language rather than augmenting one of the seven existing languages:

  - Joint Service and NATO cooperation in the specification.

  - Centralized competetive R&D for a quality product.

  - Compatibility with modern software practice and tools.

- Greater responsiveness to the unique needs of embedded computer systems, real-time control, dynamic error recovery, interface to nonstandard input-output devices, and distributed and parallel processing.

- Incorporation of state-of-the-art software and language technology to remedy the recognized deficiencies of the interim approved languages.

- Increased emphasis on software reliability, modifiability, and efficiency in the language.

- Greater portability of support software, software tools, and libraries than can be achieved from high order languages designed for a specific system or target machine.

o The DoD High Order Language Working Group is in the process of specifying a common high order computer programming language for embedded defense systems. This project has been formally under way for four years, and has produced an iterative series of language requirements and an evaluation of existing languages against those requirements. Two contractors, CII-Honeywell Bull and Intermetrics, were selected through competitive prototyping to produce alternative language designs. In May 1979, the CII-Honeywell Bull design was chosen, and a period of intensive design validation and refinement began. It is anticipated that the language specification will be complete by April 1980, and that its use in major defense systems will be authorized by a revision to DoDI 5000.31. Efforts during the succeeding twelve to eighteen months will determine the implications of phasing out some of the seven languages authorized for use under DoDI 5000.31.

Research Direction and Action

o The Services will share equally in the cost of the common language program.

o DARPA is responsible for technical management of the language specification effort, and for the development of an initial language control capability. A permanent language control agent will be designated when the language is authorized under DoDI 5000.31.

o The Services will have primary responsibility for language design validation and will cooperatively develop a common set of programming tools.

C.3 Specification of Resusable Software Functions

This area provides for R&D to identify and specify common software functions in DoD applications so that they can be implemented as standard reusable software products. In many cases, it may be possible to implement functions in firmware or hardware once their external interface is stabilized (see B.2, Hardware/Software/Firmware Tradeoffs). Examples are operating systems, data management systems, scientific subroutines, report generators and graphics interface functions. This R&D area is closely related to areas C.1 and C.2, since reusable software functions will be distributed in libraries for approved high order languages.

Problem Summary

- High cost of software.
- Unreliable software.
- Lack of transferability of software investment from one project to the next.
- Unavailability and/or lack of reuse of general purpose run-time modules such as operating systems, data management systems, telecommunications handlers, and network interface modules.

Technical Issues and Approach

- The fastest way to reduce software costs is to use existing code instead of writing new code from scratch. Embedded computer applications are lagging far behind commercial applications in the development of reusable software products. There are a variety of commercially available software products for business and scientific applications. In mature applications such as accounting, inventory control, and budget planning, a small fraction of the code to satisfy new requirements is written from scratch. Required programs are constructed out of library routines with minor modifications to satisfy idiosyncratic needs and to improve efficiency.
- DoD organizations and contractors developing software for embedded applications are aware of the enormous leverage gained from reusing software modules. Every programming team has a library of general purpose subroutines accumulated from past projects. Unfortunately, few of these private libraries have ever been documented and packaged for

distribution to outside users, primarily because of the lack of incentives for developers to invest in product development, marketing, and customer support. Prior to DoDI 5000.31, the military software market was so disorganized that there was no practical way to accumulate standard libraries for the hundreds of incompatible languages and machine architectures. The problems, however, are not entirely institutional. Significant technical problems are associated with the use of standard operating systems and library functions in demanding real-time applications.

- Private library routines used in embedded applications are often modified every time they are used, perhaps to eliminate unnecessary options, change the data types or storage representation of variables, add functional capability, etc. In some applications, an order of magnitude improvement in execution speed can be obtained with relatively little effort by someone who is intimately familiar with the internal structure and performance characteristics of library routines. In effect, the leverage comes from the programming team's thorough understanding of a particular class of algorithms rather than from their ability to utilize existing code.

- Some successful software products have eliminated the need for users to understand the internal structure of library functions by providing tools that automatically customize software. For example, in the mid 1960's, the Air Force Weapons Laboratory implemented a hydrocode calculation tool that could pull together library routines and optimize them to perform large calculations efficiently. A sophisticated implementation of this idea for small business applications is the recently announced commercial product called "Programming by Example".

- An alternative to software specialization is to provide special hardware support for standard library functions. For example, when floating point arithmetic was implemented in software, it was often appropriate to customize the implementation for a specific application. Modern computers, however, have special hardware and microcode support for floating point. This special hardware eliminates the need (and the capability) to customize the floating point implementation for specific applications. Other software products which typically depend on special hardware to improve their performance are operating systems and data management systems. The use of special hardware to improve the performance of frequently used functions will become increasingly attractive with the advent of Very Large Scale Integrated Circuits technology.

- o The new common high order language (Ada) will be a strong catalyst for the R&D of standard software products. It is clear that Ada will be widely used, and that it will endure for at least a decade. Furthermore, it will be available for all militarized computers. Hence, for the first time there is a common language in which to invest in reusable software products for embedded applications.

Research Direction and Action

- o The High Order Language Working Group will provide DoD-wide coordination of the implementation of standard applications libraries for Ada.

- o Designated language control agents will manage the standard applications libraries for the languages under their control.

C.4 Standard Instruction Sets and I/O Interfaces

This R&D area addresses the standardization of hardware/software interfaces, including central processing unit instruction sets, input-output interfaces, operating systems, and network protocols.

Problem Summary

- o Wasteful reimplementation of compilers and other development tools for new hardware.

- o Use of assembly language because no compiler for an approved language is available to produce code for the operational computer hardware.

- o The cost, schedule, and reliability risks of compiler development during the initial phases of software development projects.

- o Costly conversions of applications software to run on different hardware.

- o Inability to exploit hardware technology advances because of the cost, risk and/or time to convert existing applications software.

Technical Issues and Approach

- o Embedded systems software is especially difficult to transport to new hardware, because it must be hardware-dependent and highly optimized to satisfy real-time scheduling deadlines.

- It is useful to classify software according to the technological factors which affect portability:

  1. Software which must embody detailed knowledge of the underlying hardware (e.g., low level interfaces to hardware devices and the code generation portion of compilers).

  2. Software which embodies knowledge of the underlying hardware because of very large (order of magnitude) performance implications, even though the correct functionality could be provided in a hardware-independent fashion (e.g., operating system kernels, data management systems, and packet switching software).

  3. Software which depends intimately on software in categories 1 and 2 (e.g., applications software that depends heavily on underlying data management and operating systems software).

  4. Software written in a high order language in such a way that there are no hardware dependencies (e.g., many FORTRAN and COBOL applications programs).

- Most commercial software is in either category 3 or category 4. Nonetheless, conversion problems are widespread. These can usually be traced to the use of data management systems, and the fact that high order languages are inadequate at minimizing and isolating operating system dependencies.

- The situation with embedded software is much worse than with commercial software. Much embedded software is in category 1 and category 2. Special militarized computers tend to have idiosyncratic processor instruction sets which preclude the use of generally available software tools. Lacking an efficient compiler for an approved high order language, much embedded software is written in assembly language, and therefore must be classed as unportable.

- Various research projects are under way to improve the portability of software in categories 1 and 2. Proposed solutions are to use high order languages, isolate hardware dependencies, and to develop semiautomated tools for constructing efficient implementations of widely used software products for specific hardware.

- The alternative to conversion is software interface standardization. Key technical issues are

- Selection and rigorous specification of standard processor instruction sets and accompanying assembly level support software.

- Selection and rigorous specification of standard input/output interfaces (hardware level and device control software).

- Selection and rigorous specification of standard computer network protocols.

- Identifying applications for which current standard computers or interfaces are ill-suited, and determining whether new standards should be defined.

Research Direction and Action

- Each Service will develop a taxonomy of its applications. The Computer Resources Acquisition Panel will consolidate these taxonomies, and use them to determine the cost/performance implications of proposed standards.

- OUSDR&E (AP) will develop software interface standards with the assistance of the Computer Resources Acquisition and Instruction Set Architecture Panels of the MSC-ECR.

- OUSDR&E (C3I) has the lead in developing computer network protocol standards.

- The Services will perform required technology evaluations to support OUSDR&E.

D. COMPUTER HARDWARE

D.1 Standard Militarized Computers

This area provides for the development of standard militarized computers.

Problem Summary

- o Excessive computer resources life cycle costs.
- o Obsolete militarized hardware.

Technical Issues and Approach

- o The market for militarized computers consists largely of DoD and foreign military sales to allies.
- o Since the market for militarized computers is small relative to the market for commercial computers, R&D cost can become comparable to or exceed manufacturing costs if DoD allows the market to become too fragmented.
- o Given the rapid rate of technological progress in semi-conductor technology, DoD program managers will be driven to develop new computers to improve reliability and to reduce size, weight, and power if up-to-date militarized computers are not made available on a regular basis.
- o Software systems have a lifetime spanning several generations of computer hardware, so portability of software from one generation to the next must be a major factor in DoD's military computer hardware product line strategy.
- o Field maintenance is the other major factor in evaluating proposed military computer product line strategy.
- o The Navy's existing standard computers are the 642B, UYK-7, UYK-20, and AYK-14. There is software compatibility between the UYK-20 and AYK-14.
- o The Army uses the GYK-12, UYK-19, and UYK-41.

- The Air Force has developed the AYK-15A to be its standard avionics computer.

Research Direction and Action

- The Instruction Set Architecture Panel of the MSC-ECR will recommend the appropriate level of standardization for militarized computers. It will pay particular attention to the problem of field maintenance of military computer systems. Specifically, it will document logistics support dependencies which cross system and Service boundaries.

- The Services will support the Panel's activities with funds from this R&D area.

- Service efforts which are already under way to procure updated militarized computer hardware will continue:

  - The Navy will develop the UYK-43 and UYK-44 software-compatible follow-ons to the UYK-7 and UYK-20, as recommended in the final report of the Navy Embedded Computer Review Panel published in October 1978.

  - The Air Force will qualify two vendors to produce the AYK-15A.

  - The Army will procure a family of militarized computers based on the UYK-41 architecture, and develop a software compatible follow-on to the GYK-12.

D.2 Standard Militarized Peripherals

This R&D area provides for the specification, qualification, and in selected cases, the development of standard militarized peripherals, including disks, magnetic tapes, other mass storage devices, printers, and terminals.

Problem Summary

- Unavailability of militarized peripherals.

- Excessive computer resources life cycle costs.

- Obsolete militarized hardware.

Technical Issues and Approach

- o The market situation for militarized peripherals is similar to that for militarized computers--they are developed to satisfy unique DoD requirements, and even when companies decide to invest internal funds to develop new militarized peripheral product lines, they are inevitably targeting those products for the DoD market.

- o The total market for militarized peripherals is small compared to the market for standard commercially available peripherals. Hence, R&D costs constitute a significant fraction of per unit costs. DoD must avoid fragmenting the market further by funding the R&D of too many variants of each kind of peripheral device.

- o Convergence on a few militarized peripheral product lines will simplify field maintenance and support, and hence lower life cycle costs.

- o New militarized peripheral products must be developed or (when developed by the private sector) qualified.

- o The state of the art for computer peripherals is advancing almost as rapidly as that for processors and primary memory. Hence, DoD program managers will develop their own peripherals to achieve improved reliability and to reduce size, weight, and power if new standard militarized computer peripherals are not made available on a regular basis.

- o Standard input/output interfaces should be defined so that militarized peripherals can be interfaced to any militarized computer.

Research Direction and Action

- o OUSDR&E (AP) will establish a product line strategy for militarized peripherals and input/output interface standards with the assistance of the Computer Resources Acquisition Panel.

- o The Services will develop specific peripheral product lines for their applications in accordance with the DoD product line strategy.

D.3 High Performance Computer System Technology

This R&D area provides for the development of advanced technology for very large scale computational or scientific computers. Based upon 1979-80 available computers, "high performance" is defined as those computers with a processing data rate (PDR*) in excess of 1,000 megabits/second.

Problem Summary

- o Unavailability of high performance computers to meet unique DoD requirements.
- o High cost of high performance computers for unique DoD requirements.
- o Long development time, and hence electronic component obsolescence, for high performance computers.

Technical Issues and Approach

- o High performance computer system technology makes use of all the basic components and techniques required for small-to-medium computers, but achieves considerably greater computing speeds of "number-crunching" capacity by the allocation of more time and resources to tailoring both hardware and software to accomplish the specific tasks imposed by complex military problems. Because the demand for high performance computers to meet such unique requirements is relatively limited and not subject to normal commercial or competitive restraints of cost, size and standardization, unique management and software programs, testing routines and operating routines are applied to make the best use of the

---

*PDR - Processing data rate is the product of the "average number of bits transferred per instruction" and the "processing rate".
"Average number of bits transferred per instruction" is the sum of:
(a) the number of bits in a fixed or floating point instruction;
(b) 0.40 times the number of bits in a fixed point "operand"; and
(c) 0.15 times the number of bits in a floating point "operand".

"Processing rate" is the reciprocal of the sum of:
(a) 0.85 times the average "execution time" of a fixed point addition;
(b) 0.09 times the average "execution time" of a floating point addition; and
(c) 0.06 times the average "execution time" of a floating point multiplication.

high-performance features of the hardware.

- The demands for solving complex problems in weapons development, nuclear weapons testing and weapons targeting place unusual requirements on the whole range of hardware available to those responsible for designing and producing high performance computer systems. While such systems may not incorporate the very latest or most advanced version of every integrated circuit, memory, disk or tape drive because 3 or 4 years are now required only to design and develop the system, their software and extensive testing programs result in computers of considerably higher PDR than smaller ones using the very latest components. A processing data rate of 1,000 mega-bits/second is roughly the state-of-the-art for the highest performance commercial computers circa 1979. Not only are advanced circuits used, but overall architecture developments involve many more logic circuits than do other computers, and extensive use of parallelism and vector processors provides additional means of achieving greater capabilities.

- The primary requirements by the military for high performance computer systems are for solving and dealing with complex problems associated with weapons development. While greater use will be made of small-to-medium computers incorporated into computer networks, certain types of problems will always exist in both the military and scientific/industrial sectors for which solutions will require unique capabilities available only on such high performance computer systems. The Navy, for example, requires massive data processing to deal with ocean traffic surveillance and anti-submarine warfare missions.

- High performance computer systems, such as CRAY, already may exceed a PDR of 4,000 megabits/second and the largest computers of CDC, IBM and Fujitsu are approaching or possibly just exceeding 1,000 megabits/second. Large, moving-head computer disc systems will make possible large, on-line data bases with data manipulation and convenience in operation that are qualitatively different from present generation high performance computer systems. A new design approach demonstrated by Lawrence Livermore Laboratory in the S1 project uses a unique computerized design system to substantially reduce the elapsed time and number of man-hours required to develop large high performance machines. Josephson junction circuit technology, which is based on supercooled semiconductor materials, may make possible the development of computers with processing data rates approaching a billion bits per second.

Research Direction and Action

- o The Navy will develop and test a 4x4 configuration of S1 processors.

- o The Navy will determine the extent to which the computer-aided design and manufacturing technology used to develop the S1 computer at Lawrence Livermore Laboratory changes the economics of custom computer design for applications requiring very high performance. This evaluation will be an important input to the Acquisition Panel.

- o A consortium of DoD user organizations will fund the exploratory R&D of Josephson junction technology, and review other promising technologies for the R&D of very high performance computers.

- o Additional exploratory development efforts will be funded as promising technologies emerge.

APPENDIX A

CHARTER FOR THE
R&D TECHNOLOGY PANEL

adopted by the

Management Steering Committee for Embedded
Computer Resources

29 October 1976

MEMORANDUM OF UNDERSTANDING

The Department of Defense Automation Objectives dated March 25, 1976 included an initiative to "outline a program of applied research requirements." The intent of this particular initiative was to establish a formal process that would provide for the production of a coordinated set of R&D objectives and supporting projects to accomplish these objectives in the area of general purpose data processing.

Similarly, during the same period of time DoD Directive 5000.29, dated April 26, 1976, established a mechanism for resolving many problems associated with the management of computer resources in major Defense systems. In addition to addressing other problems, the Management Steering Committee for Embedded Computer Resources (MSC-ECR) established the need for a coordinated approach to solving the R&D problems associated with computer resources in major Defense systems (i.e., embedded computers). Hence, the R&D Coordinating Panel was one of the four panels to be established under the MSC-ECR.

The computer science problems that plague the general purpose area are very similar to those that plague the area of embedded computers. Therefore, a single panel supporting both communities seems highly appropriate. Moreover, since the ODDR&E must review the computer science R&D of both communities this panel would provide the proper mechanism for establishing and maintaining a unified and cohesive R&D program. Hence, panel efforts would be supported by both the ADP Policy Committee representing the general purpose area and the MSC-ECR representing the area of embedded computers.

An approved charter for this panel is attached.

Representing Embedded Computer Systems Area

Barry C DeRoze

OASD(I&L) Barry DeRoze

Representing General Purpose ADP Area

John Carabello

OASD(C) John Carabello

Tom Nyman

ODDR&E Tom Nyman

CHARTER

FOR THE

RESEARCH AND DEVELOPMENT TECHNOLOGY PANEL

1.0 OBJECTIVES

The objectives of the Research and Development Technology Panel (RDTP) are to:

a) provide a coordinated research and development program plan to supply the technology base which supports all computer resource applications within DoD;

b) provide recommendations and advice to both the Management Steering Committee for Embedded Computer Resources and the ADP Policy Committee to avoid unproductive overlap, gaps, or duplication of effort in the conduct of DoD's computer resources research and development efforts;

c) formulate and as necessary propose additions and deletions to computer resource R&D objectives for joint consideration by the MSC-ECR and the ADP Policy Committee;

d) serve as a forum for coordinating technology investment strategy for the Military Departments and Defense Agencies;

e) review R&D programs to monitor progress toward established objectives; provide annual progress appraisal against each established objective, jointly to the MSC-ECR and ADP Policy Committee;

f) identify technologies which appear ready for operational use, and assist the MSC-ECR, DDR&E and the ADP Policy Committee in conducting and evaluating suitable demonstrations;

g) provide technical comments on Technology Annexes to DCPs and PMs as requested by the MSC-ECR.

h) assist Program Managers and System Project Offices in the identification of technology deficient areas and in promoting technology transfer.

In pursuing the above objectives, the scope of the RDTP will encompass all computer resource research and development activities within the Military Departments and Defense Agencies, and will include both embedded computer resources and general purpose automatic data processing application areas.

2.0 REFERENCE

The RDTP functions in accord with the policies of DoD Directive 5000.29, "Management of Computer Resources in Major Defense Systems," 26 April 1976 and DoD Directive 5100.40, "Responsibilities for the Administration of the DoD ADP Program," 19 August 1975.

3.0 CHAIRMANSHIP

The RDTP shall have a permanent Chairman selected by, and representing the Director, Defense Research and Engineering. The Chairman will be the responsible spokesman for the RDTP, and will administer the Panel affairs.

4.0 MEMBERSHIP

The membership of the RDTP shall be composed of not more than three representatives from each Military Department and Defense Agency. Members of the RDTP shall be selected by their respective DoD Component and their scope shall represent both embedded computer resources and general purpose automatic data processing application areas.

5.0 ACTIVITIES

In fulfilling the objectives of the Charter, the RDTP shall as a minimum carry out the following activities:

a) develop, propose, and maintain a DoD Computer Resource R&D Technology Program Plan

   1) develop computer resources technology objectives,

   2) identify current effort devoted to these objectives,

   3) identify and prioritize critical areas which need immediate emphasis,

   4) plan near, mid, and long term solutions to each objective,

5) identify and recommend responsible agency or joint activity to lead on areas of common interest,

6) identify resource implications of these efforts;

b) meet at least quarterly to discuss progress toward objectives;

c) prepare and present an annual report on R&D Technology Progress to a joint meeting of the MSC-ECR and the ADP Policy Committee.

d) Provide summary briefings on Panel activities at each formal meeting of the MSC-ECR;

e) carry out specific tasks as directed by the Chairman of the MSC-ECR and DDR&E. The Chairman of the ADP Policy Committee will request specific tasks through his participation on the MSC-ECR.

APPENDIX B

SOFTWARE SCIENCE AND TECHNOLOGY BASE TECHNOLOGY OBJECTIVES

adopted by the

Management Steering Committee for Embedded Computer Resources

31 August 1976

SOFTWARE SCIENCE AND TECHNOLOGY BASE

APPROVED TECHNOLOGY OBJECTIVES

The Military Departments, through the Research and Development Technology Panel to the Management Steering Committee for Embedded Computer Resources (MSC-ECR), have formulated broad technology objectives for evaluating the software technology base. These objectives reflect current deficiencies in both embedded and general purpose computer application areas. The objectives were proposed to the MSC-ECR and subsequently adopted (with minor changes) in September 1976. The technical objectives as adopted are itemized below. No efforts to prioritize among the objectives have been made.

1. Project Management:

    1.1 Resolve technical issues associated with the preparation of life cycle computer resources.

    1.2 Develop improved methods and tools for planning, estimating and controlling software development.

    1.3 Develop criteria and procedures for configuration item definition, interface definition and control, and change control and impact assessment of changes.

    1.4 Develop methods, languages and tools for describing and validating requirements.

    1.5 Establish risk analysis techniques to minimize unforeseen cost and schedule impacts from system requirements.

    1.6 Develop qualitative and quantitative measures of software quality.

    1.7 Establish a uniform software error and cost data collection and analysis methodology.

    1.8 Perform computer technology assessments and develop techniques for measuring the impact of software technology advances on productivity.

    1.9 Demonstrate new technology concepts through prototype or experimental proofing prior to full scale technology transfer to on-going system applications.

2. System Architecture

    2.1 Develop concepts in computer system architecture which reduce software costs, improve timeliness, increase quality, and/or enhance man-maching interaction.

2.2 Develop software techniques which increase the usefulness of computer architectures.

2.3 Develop methods for designing computer systems which explicitly consider the trade-offs between hardware, software and firmware.

2.4 Develop and demonstrate techniques and concepts to ensure security of information systems.

2.5 Demonstrate techniques for flexible, interoperable, and reliable data management systems.

3. Programming Environment

3.1 Identify properties of programming languages and compilers which provide for effective control of software development, enhanced quality, and reduced cost.

3.2 Develop tools which automate the clerical aspects of software design and synthesis.

3.3 Develop methods and tools for testing which allow determination of whether adherence to the requirements has been achieved within a stated tolerance, or which otherwise quantify reliability.

3.4 Develop techniques and supporting tools for proving that programs and specifications are consistent.

3.5 Demonstrate techniques and tools which enhance the maintainability and modifiability of software.

3.6 Demonstrate techniques and tools for software transportability which significantly reduce the effort to modify software so it will execute on different computer hardware.

3.7 Develop software engineering methods which exploit new tools to improve the quality of software and provide for effective control of development.

3.8 Develop programming environments to facilitate the flexible use of many tools in combination with each other, and the addition of new tools.

4. Reusable Software and Tool Availability

4.1 Develop techniqes for formal specification of standard software products.

4.2 Develop technology for adapting standard software products to specific applications, and for cost effectively maintaining the resultant product families.

4.3 Demonstrate techniques for efficiently transporting standard products to different hardware.

4.4 Establish language control facilities and develop necessary supporting tools.

4.5 Eliminate the need to build new versions of software tools just to make them available for new languages and different computer systems.

4.6 Establish easily accessible repositories and distribution systems for software tools and other reusable software.

4.7 Investigate the consolidation of the many HOLS in common use to a smaller number of common HOLS.

APPENDIX C

ANNUAL REPORT OF THE DEFENSE COMPUTER RESOURCES
TECHNOLOGY PROGRAM - FY 1978

8 June 1979

Annual Report of the Defense Computer Resources Technology Program - FY 1978

SUMMARY AND CONCLUSIONS

The *Defense System Software R&D Technology Plan* published in September 1977 outlined key technology issues for DoD embedded computer applications and established target budgets for R&D initiatives addressing those issues. This report summarizes accomplishments to date in the program and funding plans for FY1979 through FY1981. It also identifies managerial issues requiring special emphasis.

The second edition of the Technology Plan was begun during 1978. The most significant change is the expansion of the Plan's scope to include military computer hardware. The Plan's outline was also revised to regroup efforts in a more useful manner, and to clarify the intent of some subdivisions. All parts of the text were considerably extended and expanded to provide more comprehensive descriptions of problems, issues, and R&D directions.

Major accomplishments during FY1978 include the establishment of language control facilities for approved languages, the completion of requirements definition and preliminary design for the new common high order language (Ada), the achievement of an initial operating capability for the National Software Works on the ARPANET, the completion of guidebooks for command and control system acquisition, and several successful applications of the Computer Aided Design and Specification Analysis Tool (CADSAT). The Navy Embedded Computer Review Panel completed an evaluation of Navy computer resources requirements, and recommended the immediate development of software-compatible follow-ons to the AN/UYK-7 and AN/UYK-20. The Army completed a preliminary form fit and function specification for the Military Computer Family of software-compatible computers. A unique computer-aided design technique was developed which allowed a prototype high-performance processor (the S-1), with a throughput in excess of ten million instructions per second, to be developed with only 2 to 3 man-years of effort.

Key areas for which funding deficiencies still exist include error-resistant systems, standardization of input/output and network interfaces, and specification of standard reusable software functions. Increased DoD-wide coordination of hardware technology initiatives is needed. Proven militarized hardware incorporating state of the art semiconductor technology must be made available at reasonable cost, and with adequate provision for software portability and logistics support.

Efforts are building up and becoming better focused, but present budgets are sadly inadequate to allow meaningful efforts to be formed in some areas. Funding during FY1978 was about 60% of the minimum threshold requested by the Management Steering Committee for Embedded Computer Resources. During FY1979, software-related activities (categories A through C of the Plan) will be funded at about 80% of the desired level, and there will be a shortfall of as much as $10M in the military hardware area. A similar pattern seems to be developing for FY1980.

Tables 1-13, beginning on page 8, present FY1978-FY1981 budget data by technology area, Service, and program element, as submitted to OUSDR&E (R&AT) in January 1979.

A. LIFE CYCLE MANAGEMENT TOOLS - HIGHLIGHTS

Funding	FY78	FY79	FY80	FY81
Army	1.6	2.5	3.0	3.1
Navy	0	0	0	0
Air Force	1.3	3.7	2.8	4.0
DARPA	0	0	0	0
	2.9	6.2	5.8	7.1

Progress to date

Requirements Analysis: The Air Force and Army have surveyed existing requirements definition tools and techniques. The Air Force selected one of these techniques and refined it to improve its suitability for Air Force applications. The result is the Computer Aided Design and Specification Analysis Tool (CADSAT) which can execute on three different host computers (the Honeywell H6180, the IBM 370, and the Univac UYK-7). It has been used successfully on the GEODSS (Ground Electro Optical Deep Space Surveillance) command and control system development, for the Interim Upper Stage portion of the Space Shuttle, and by the Navy for the NTDS restructure. The Army is developing a complementary requirements analysis technology called "System Sketching" to provide early feedback about system characteristics to prospective users.

Cost/Schedule/Quality Data: The Air Force has established a Software Data Analysis Center at RADC. Data has been acquired for the PAVE PAWS and SAMTEC systems and placed in that repository. The Air Force has completed a study to identify factors having an adverse impact on software cost.

Metrics and Planning: The Army has developed a cost estimation and sizing model. Although it was originally aimed at decision support applications (command and control and management information systems), initial experiments indicate it has wider applicability. The Air Force has evaluated the Army model and has also applied a commercial model (PRICE-S) to avionics applications. The Air Force has developed a computer hardware estimation model.

Configuration Management: The Air Force has begun evaluating the use of CADSAT for change impact assessment. The Army has surveyed the commercial marketplace, and acquired a minicomputer based configuration management system to conduct an experiment in post-deployment software support.

Guidance Documents: The Air Force completed the 16 volume command, control and communications series, which has been distributed to more than 500 user groups throughout DoD. The Army conducted a Life Cycle Management Workshop.

Prognosis

- o The software development process is still poorly understood. Guidelines for managing the interaction of requirements analysis, design, risk assessment, and risk-reducing experimental development are inadequate. Cost estimation models are just becoming available.

- o A model of the complete software development and maintenance process is needed which deals explicitly with risk assessment and risk reduction, requirements evolution, and technological change. Such a model must be tested in key DoD applications, and applications dependencies identified. Tools for requirements analysis, planning, cost estimation, design specification and configuration control must be tied together in a way which is consistent with the process model. The Air Force and Army plan to address these issues, so no management action is required at this time.

B. SYSTEM DESIGN AND ARCHITECTURE - HIGHLIGHTS

Funding	FY78	FY79	FY80	FY81
Army	2.4	3.6	5.5	8.0
Navy	.6	.8	.9	2.8
Air Force	5.7	6.2	7.0	9.7
DARPA	3.3	2.6	2.8	3.0
	12.0	13.2	16.2	23.5

Progress to date

Error-Resistant Systems: DARPA has completed concept demonstrations of formal verification technology. Experimental applications have been initiated to test the robustness of the technology, and to determine how to use a verification system to identify effective error detection and recovery logic for implementation in hardware. Applications include verification of a secure operating system kernel and verification of data management system reliability.

Hardware/Software/Firmware Tradeoffs: The Navy has developed a Performance Oriented Design (POD) System for predicting the performance of systems with multiple processors and other active devices. POD will be evaluated in a command and control application during the coming year. The Air Force is assembling an integrated system of analysis tools for making hardware/software/firmware tradeoffs. They have identified software criteria to be used in selecting microprocessors for qualification via MIL-STD-38510.

Distributed Systems: The Air Force and DARPA have implemented an initial version of the National Software Works on the ARPANET. It provides software tool interoperability among the IBM 360, Honeywell 6180 and DEC PDP-10. Included are protocols for file interchange and format conversion, interprocess communication, a common file system, and basic configuration management aids. This system will be used for distributed software development and configuration management experiments involving the Air Force Logistics Command, Naval Ocean Systems Center, and other military users.

Multilevel Computer Security: DARPA has completed a secure minicomputer concept demonstration. DARPA/DCA/NSA are jointly developing a kernelized secure operating system for the PDP-11 that incorporates technology from the initial concept demonstration, and which is sufficiently robust for use in operational demonstrations and experiments. A design for a large scale virtual machine operating system kernel (KVM) is complete and development is under way.

Testbeds: The Air Force Avionic System Analysis and Integration Laboratory (AVSAIL) was used to perform independent validation of F-16 operational flight programs, and the test design and associated software was directly transitioned to AFLC for use in the F-16 support facility. The Army's Teleprocessing Design Center has been used for TOS-2 and TACFIRE performance analysis, and to prove the practicality of software portability using microcode emulation.

Prognosis

A variety of promising technology development efforts are under way. The new common high order language (Ada) will be a direct mechanism for transferring new capabilities into operational use as they are developed. For example, tools for implementing error-resistant software will be implemented in Ada. Standard distributed system interface software will also be implemented in Ada and distributed as part of the Ada library. Increased Service funding is needed in the area of error- resistant systems.

C. SOFTWARE PRODUCT SPECIFICATION AND STANDARDIZATION - HIGHLIGHTS

Funding	FY78	FY79	FY80	FY81
Army	2.0	2.5	4.4	3.9
Navy	.2	1.0	1.2	1.5
Air Force	1.1	3.0	3.2	4.3
DARPA	0	0	0	0
	3.3	6.5	8.8	9.7

Progress to date

Support of Standard Languages: The Air Force has established an R&D prototype J73 control facility and an operational capability for J-3 control. The Navy has an established capability for CMS-2 control, and an SPL-1 control capability is under development. The Air Force has developed an integrated set of software tools for the JOVIAL language based on the programming support library concept. It will be used by WWMCCS, SAC, DMA and PAVE PAWS.

New Standard Language: Requirements have been defined for a common high order language suitable for U.S. and NATO language convergence. Four contractors developed preliminary designs to those requirements. Two were selected to spend one year refining their designs and deliver complete language specifications in March 1979. The chosen language will be matured during an intensive design validation period from May 1978 to December 1979.

Reusable Software Functions: This area is defined for the first time in the R&D plan.

Standard Instruction Sets/Interfaces: The Army has completed an evaluation of existing computer instruction sets. They have also done a life cycle cost analysis of existing instruction sets against a standard (AN/UYK-41) instruction set. The Navy Embedded Computer Review Panel evaluated alternatives for supporting, improving and/or replacing the AN/UYK-7 and AN/UYK-20 computers. The Air Force completed specification of the MIL-STD-1750 avionics computer architecture, which will be used for the F-111 A&E update program.

Prognosis

The DoD high order language standardization program is on schedule, and momentum is building. DoD needs some level of processor instruction set standardization, so the criticisms of draft 5000.xx must be understood and a new, acceptable DoD instruction drafted. The benchmark programs used to select the PDP-11 for the Military Computer Family program must be reviewed by all three Services to be sure that they are representative of their specific applications. An analysis of variance must be performed to determine the minimal collection of instruction sets which meet DoD's needs. Increased emphasis must be given to standardization of input/output and network interfaces and to the new area of Reusable Software Functions.

D. COMPUTER HARDWARE - HIGHLIGHTS

Funding	FY78	FY79	FY80	FY81
Army	2.0	3.9	12.0	14.5
Navy	.6	1.9	5.0	12.0
Air Force	0	2.0	2.0	0
DARPA	0	0	0	0
	2.6	7.8	19.0	26.5

Progress to date

Military Computer Family: The Army has completed a preliminary form fit and function specification for the system and subsystems of a family of software compatible computers suitable for multi-source procurement.

S-1: The Navy has developed advanced computer-aided design techniques and demonstrated their power by developing a prototype high-performance processor with throughput in excess of ten million instructions per second.

MIL-STD 1750: The Air Force has developed a standard instruction set architecture for avionics systems.

Prognosis

Increased DoD-wide coordination of hardware technology initiatives is needed. Proven militarized hardware incorporating state-of-the-art semiconductor technology must be made available at reasonable cost, and with adequate provision for industry competition. DoD's acquisition strategy for militarized computer hardware must address software portability and logistics support issues. Other unique military computer requirements, such as signal processors, also need to be addressed. The MSC-ECR will form an Instruction Set Architecture Panel to give high-priority attention to these issues during the coming year.

TABLE 1

DEFENSE COMPUTER RESOURCES TECHNOLOGY PROGRAMS

($ millions)

	FY 78	FY 79	FY 80	FY 81
A. LIFE CYCLE MANAGEMENT TOOLS				
A.1 Requirements Analysis	1.1	1.8	1.8	2.5
A.2 Cost/Quality Data Collection and Analysis	.8	1.1	.8	1.3
A.3 Metrics and Planning Technology	.6	.8	.9	.8
A.4 Specification, Control and Configuration Management	.2	1.2	1.4	1.4
A.5 Policy and Procedure Guidance Documents	.2	1.3	.9	1.1
AREA TOTAL	2.9	6.2	5.8	7.1
B. SYSTEM DESIGN AND ARCHITECTURE				
B.1 Error Resistant Systems	.3	.3	.5	.8
B.2 Hardware/Software/Firmware Tradeoffs	1.9	2.5	2.6	3.6
B.3 Distributed Systems	3.6	3.1	4.1	4.9
B.4 Multilevel Computer Security	1.6	1.8	2.6	5.1
B.5 Applications Testbeds and Experimental Facilities	4.6	5.5	6.4	9.1
AREA TOTAL	12.0	13.2	16.2	23.5
C. SOFTWARE PRODUCT STANDARDIZATION				
C.1 Support of Present Standard Languages	1.6	3.0	2.9	2.4
C.2 New Standard Language Development (Ada)	.7	2.2	4.3	5.3
C.3 Specifications of Reusable Software Functions	.3	.5	.5	.6
C.4 Standard Instruction Sets and I/O Interfaces	.7	.8	1.1	1.4
AREA TOTAL	3.3	6.5	8.8	9.7
D. HARDWARE				
D.1 Standard Military Computers	2.0	5.6	14.0	18.5
D.2 Standard Military Peripherals		.7	4.0	6.0
D.3 High Performance Computers	.6	1.5	1.0	2.0
AREA TOTAL	2.6	7.8	19.0	26.5
TOTALS	21.0	33.7	49.8	66.8

8

DTIC